Necessary and sufficient condition for unique flows in certain intersection models

Gunnar Flötteröd[1]

May 12, 2011

[1] in the context of joint work with R. Corthout, C. Tampere, F. Viti

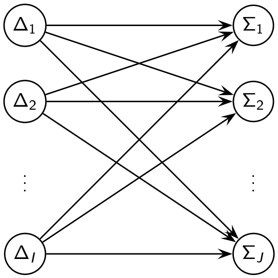
Motivation

- deterministic dynamic traffic assignment
- ambiguous solutions at network level possible
- what about ambiguous solutions at *node* level?

Outline

Outline

Modeling assumptions (1/2)



- Δ_i is flow demand of ingoing link i
- Σ_j is flow supply of outgoing link j
- not every i needs to compete for every j
- also covers internal conflicts

Modeling assumptions (2/2)

- every ingoing link i maximizes its inflow q_i^{in} subject to
  - demand constraints $q_i^{\mathrm{in}} \leq \Delta_i$
  - supply constraints $q_j^{\mathrm{out}} \leq \Sigma_j$ on node outflows q_j^{out}
  - first-in/first-out and flow conservation: $q_j^{\mathrm{out}} = \sum_i \beta_{ij} q_i^{\mathrm{in}}$ with turning fractions β_{ij}
  - q_{i_1} and q_{i_2} are constrained by $j \Rightarrow q_{i_1}/q_{i_2} = \alpha_{i_1 j}/\alpha_{i_2 j}$ with <u>strictly positive and finite priorities α_{ij}</u>
- priorities are decisive for unique flow solutions

Outline

Modeling assumptions

Sufficient condition

Necessary condition

Summary and outlook

Sufficient condition and solution algorithm

- *Flows are unique if there are positive α_i and c_j such that*

$$\alpha_{ij} = \alpha_i c_j$$

holds for all upstream links i that could enter constraint j.

- Proof: By known solution algorithm.
  1. assign unique inflow priority $\alpha_i = \alpha_{ij}/c_j$ to every ingoing link
  2. set all inflows to zero; label all $i = 1 \dots I$ as "unconstrained"
  3. while there are unconstrained inflows left:
     - 3.1 increase all unconstrained inflows proportionally to their priorities until the next constraint binds
     - 3.2 label all inflows that reached the constraint as "constrained"

Outline

Necessary condition

- *The sufficient condition is also necessary:*
  There are positive α_i *and* c_j *such that*

$$\alpha_{ij} = \alpha_i c_j$$

  holds for all upstream links i *that could enter constraint* j.
- Sketch of proof:
  - Assume that the necessary condition does not hold.
  - Show that a non-unique solution can always be constructed.

TRANSP-OR

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Preliminaries

- *Every outgoing link j with only one ingoing link i can be transformed into a demand constraint on i.*
- Hence, for all affected i and j:
    1. replace Δ_i by $\min\{\Delta_i, \Sigma_j/\beta_{ij}\}$
    2. remove j from consideration

Test priority

- *A test priority* $\tilde{\alpha}(j_1, \ldots, j_J)$ *is defined as follows:*

$$\tilde{\alpha}(j_1, \ldots, j_I) = \begin{pmatrix} \alpha_{1j_1} \\ \vdots \\ \alpha_{Ij_I} \end{pmatrix} \quad j_1, \ldots j_I \text{ arbitrary}$$

- That is, go through all ingoing links $i = 1 \ldots I$ and assign to it one of its priorities α_{ij}.

TRANSP-OR

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Linearly dependent test prios $\Leftrightarrow$ necessary cond.

- *The necessary condition is equivalent to linear dependence of all test priorities:*

$$\alpha_{ij} = \alpha_i c_j \;\forall i,j \quad \Leftrightarrow \quad \tilde{\boldsymbol{\alpha}}(j_1,\ldots,j_I) \propto \tilde{\boldsymbol{\alpha}}(l_1,\ldots,l_I) \;\forall j_i, l_i$$

- Proof: "$\Rightarrow$" by insertion; "$\Leftarrow$" by rearrangement.
- Implication: If the necessary condition does not hold,
  - there are linearly independent test priorities $\tilde{\boldsymbol{\alpha}}^{\mathsf{A}}$ and $\tilde{\boldsymbol{\alpha}}^{\mathsf{B}}$;
  - there are ingoing links $i_1 \neq i_2$ with $\tilde{\alpha}^{\mathsf{A}}_{i_1}/\tilde{\alpha}^{\mathsf{A}}_{i_2} \neq \tilde{\alpha}^{\mathsf{B}}_{i_1}/\tilde{\alpha}^{\mathsf{B}}_{i_2}$.
- Focus on these links.

Setting of boundary conditions

- Consider two supply constraints:

$$\Sigma_{j1} \geq \beta_{i1,j1} q_{i1} + \beta_{i2,j1} q_{i2} + \sum_{l \neq i1,i2} \beta_{l,j1} q_l$$
$$\Sigma_{j2} \geq \beta_{i1,j2} q_{i1} + \beta_{i2,j2} q_{i2} + \sum_{l \neq i1,i2} \beta_{j,j2} q_l.$$

- Scale down Σ_{j_1}, Σ_{j_2}, and all Δ_i until all other Σ_j never bind.
- Ensure that i_1, i_2 are the first inflows to be constrained.

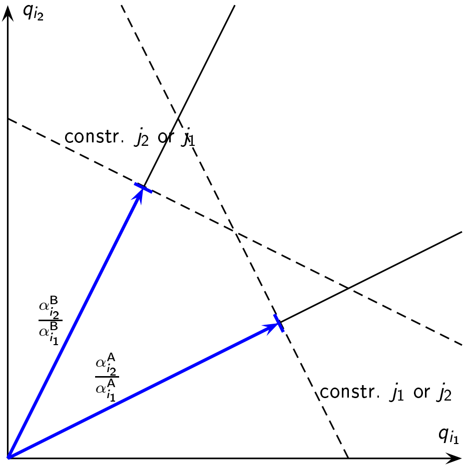
TRANSP-OR

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Three cases

1. both i_1 and i_2 compete for both j_1 and j_2
2. i_1 does not compete for j_2
3. i_1 does not compete for j_1 and i_2 does not compete for j_2

Case 1: both i compete for both j



Construction of case 1

- binding constraints:

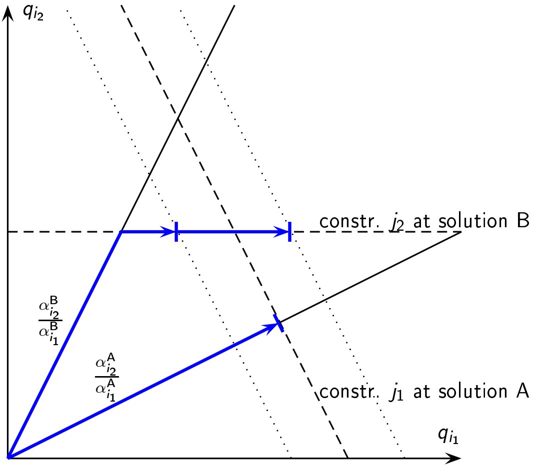
$$q_{i2} = \frac{1}{\beta_{i2,j1}}\left(\Sigma_{j1} - \sum_{l \neq i1,i2} \beta_{l,j1} q_l\right) - \frac{\beta_{i1,j1}}{\beta_{i2,j1}} q_{i1}$$

$$q_{i2} = \frac{1}{\beta_{i2,j2}}\left(\Sigma_{j2} - \sum_{l \neq i1,i2} \beta_{j,j2} q_l\right) - \frac{\beta_{i1,j2}}{\beta_{i2,j2}} q_{i1}$$

- construction:

1. set constraint slopes with the βs
2. shift constraints with the Σs

Case 2: i_1 does not compete for j_2



Construction of case 2

- bindung constraints:

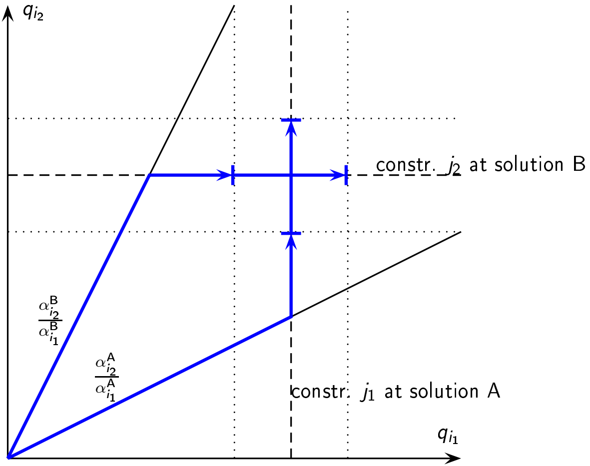
$$q_{i2} = \frac{1}{\beta_{i2,j1}}\left(\Sigma_{j1} - \sum_{l \neq i1,i2} \beta_{l,j1} q_l\right) - \frac{\beta_{i1,j1}}{\beta_{i2,j1}} q_{i1}$$

$$q_{i2} = \frac{1}{\beta_{i2,j2}}\left(\Sigma_{j2} - \sum_{l \neq i1,i2} \beta_{j,j2} q_l\right)$$

- construction:

1. set the β for some slope of j_1
2. set the Σ such that j_2 is "atop" of j_1

Case 3: i_1 (i_2) competes only for j_1 (j_2)



Construction of case 3

- binding constraints:

$$q_{i1} = \frac{1}{\beta_{i1,j1}} \left(\Sigma_{j1} - \sum_{l \neq i1,i2} \beta_{l,j1} q_l \right)$$

$$q_{i2} = \frac{1}{\beta_{i2,j2}} \left(\Sigma_{j2} - \sum_{l \neq i1,i2} \beta_{j,j2} q_l \right)$$

- construction:
  1. set the Σ somehow
  2. avoid unique solution by reducing some Δ_l, $l \neq i_1, i_2$ until it shifts a binding constraint

TRANSP-OR

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Outline

Modeling assumptions

Sufficient condition

Necessary condition

Summary and outlook

Summary and outlook

- summary: better keep the node model simple
- outlook: nonlinear internal node constraints